Sreram K *Backend developer*

sreramk26@gmail.com

+91 9369672491

https://www.linkedin.com/in/k-sreram-a04a90b7/

https://sreramk.medium.com/

https://github.com/sreramk

https://stackoverflow.com/users/3633270/sreram?tab=profile

India

EDUCATION

BE - Computer Science and Engineering, 2014 – 2019
Easwari Engineering College

PROJECTS/ACHIEVEMENTS

A machine learning library for a fully connected neural network built from scratch 2016 – 2017
Improved training performance by 0%-90% based on the sparsity. This is done by completely eliminating the need to use sparse matrices (contrary to most other frameworks) for transformation. I did this by designing the network as a graph of node objects instead of using matrices.

A key value pair database engine built from scratch 2016 – 2017
Improves disk usage by 20%-50% compared with B+ trees. Introduced random access to data records which significantly reduced local RAM usage and saved bandwidth. Implemented the storage structure as a balanced binary search tree.

Image super resolution with CNN 2019 – 2019
Gained an improvement in training accuracy by more than 90%, upon introducing a unique loss function.

A sudoku puzzle solver 2015 – 2016
Intelligently switches between using brute force and a rule based method, improving performance by 1100% (x11 times) on problems particularly designed to be hard for computers to solve. Solved all tested problems in less than 1s.

A classification algorithm built to individually map dimension subsets with its class labels 2016 – 2017
Eliminated the requirement of high-end expensive hardware to perform the training. Instead I had built this to support online training by dynamically associating the data with its class as they arrive.

Working on designing and building a conversational engine that is horizontally scalable. 2017 – present
I've designed the conversational engine to be cost effective by not requiring dedicated training resources.

SKILLS

Diverse set of languages ●●●●○
C, C++, Java, Python, JavaScript, Golang

Design stateless horizontally scalable microservices ●●●●○
With caches when needed.

Design for high availability and partition-tolerance ●●●●○
I also find ways to decouple DB from services

Read and comprehend research work ●●●○○
also related to ML and DS